

**Amendments to the Claims**

Please cancel claims 3, 11 and 19. Please amend Claims 1, 9, and 17. The Claim Listing below will replace all prior versions of the claims in the application:

**Claim Listing**

1. (Currently Amended) A switch comprising:
  - a plurality of reserved pool pools of buffers in a shared memory, each the reserved pool of buffers associated with one of a plurality of egress ports and reserved for storing data to be forwarded to an the egress port;
  - a shared pool of buffers in the shared memory, the shared pool of buffers shared by a for storing data to be forwarded to any of the plurality of egress ports;
  - a multicast pool of buffers in the shared memory reserved for storing IP Multicast packets to be forwarded to at least one egress port to members of the IP Multicast group;
  - and
  - a pool select logic which selects a free buffer to allocate from the reserved pool for storing data received from an ingress port to be forwarded to the egress port, and deallocates the selected buffer after the data has been forwarded to the egress port.
2. (Original) A switch as claimed in Claim 1 wherein the pool select logic selects a free buffer in the shared pool upon detecting no free buffer in the reserved pool.
3. Canceled.
4. (Original) A switch as claimed in Claim 3 wherein the pool select logic selects a free buffer from the multicast pool upon detecting an IP Multicast data packet received from an ingress port.
5. (Original) A switch as claimed in Claim 3 further comprising:
  - a total free counter storing the number of free buffers in shared memory, the total free counter preset to the total number of buffers in the shared memory wherein the sum

of the buffers in the multicast pool, the reserved pool and the shared pool is greater than the total number of buffers in the shared memory.

6. (Original) A switch as claimed in Claim 3 further comprising:  
 a total free counter storing the number of free buffers in shared memory, the total free counter preset to the total number of buffers in the shared memory wherein the sum of the buffers in the multicast pool, the reserved pool and the shared pool is equal to the total number of buffers in the shared memory.
7. (Original) A switch as claimed in Claim 5 wherein the pool select logic determines the availability of a free buffer in the shared memory dependent on the number of free buffers in the shared memory stored in the total free counter.
8. (Original) A switch as claimed in Claim 6 wherein the pool select logic determines the availability of a free buffer in the shared memory dependent on the number of free buffers in the shared memory stored in the total free counter.
9. (Currently Amended) A switch comprising:  
 a plurality of reserved pool pools of buffers in a shared memory, the each reserved pool of buffers associated with one of a plurality of egress ports and reserved for storing data to be forwarded to an the egress port;  
 a shared pool of buffers in the shared memory, the shared pool of buffers ~~shared by a~~ for storing data to be forwarded to any of the plurality of egress ports;  
 a multicast pool of buffers in the shared memory reserved for storing IP Multicast packets to be forwarded to at least one egress port to members of the IP Multicast group;  
 means for selecting a free buffer to allocate from the reserved pool for storing data received from an ingress port to be forwarded to the egress port; and  
 means for deallocating the selected buffer after the data has been forwarded.

10. (Original) A switch as claimed in Claim 9 wherein the means for selecting selects a free buffer in the shared pool upon detecting no free buffer in the reserved pool.
11. Canceled.
12. (Original) A switch as claimed in Claim 11 wherein the means for selecting selects a free buffer from the multicast pool upon detecting an IP Multicast data packet received from an ingress port.
13. (Original) A switch as claimed in Claim 12 further comprising:  
means for counting the number of free buffers in shared memory, the means for counting preset to the total number of buffers in the shared memory wherein the sum of the buffers in the multicast pool, the reserved pool and the shared pool is greater than the total number of buffers in the shared memory.
14. (Original) A switch as claimed in Claim 12 further comprising:  
means for counting the number of free buffers in shared memory, the means for counting preset to the total number of buffers in the shared memory wherein the sum of the buffers in the multicast pool, the reserved pool and the shared pool is equal to the total number of buffers in the shared memory.
15. (Original) A switch as claimed in Claim 13 wherein the means for selecting a free buffer determines the availability of a free buffer in the shared memory dependent on the number of free buffers in the shared memory stored in the means for counting.
16. (Original) A switch as claimed in Claim 14 wherein the means for selecting a free buffer determines the availability of a free buffer in the shared memory dependent on the number of free buffers in the shared memory stored in the means for counting.

17. (Currently Amended) A method for managing a shared memory in a switch comprising the steps of:

providing a plurality of reserved pool pools of buffers in the shared memory, ~~the~~ each reserved pool of buffers associated with one of a plurality of egress ports and reserved for storing data to be forwarded to an the egress port;

providing a shared pool of buffers in the shared memory, the shared pool of buffers ~~shared by~~ for storing data to be forwarded to any of the a plurality of egress ports port;

providing a multicast pool of buffers in the shared memory reserved for storing IP Multicast packets to be forwarded to at least one egress port to members of the IP Multicast group;

selecting a free buffer to allocate from the reserved pool for storing data received from an ingress port to be forwarded to the egress port; and

deallocating the selected buffer after the data has been forwarded to the egress port.

18. (Original) A method as claimed in Claim 17 wherein the step of selecting selects a free buffer in the shared pool upon detecting no free buffer in the reserved pool.

19. Canceled.

20. (Original) A method as claimed in Claim 19 wherein the step of selecting selects a free buffer from the multicast pool upon detecting an IP Multicast data packet received from an ingress port.

21. (Original) A switch as claimed in Claim 20 further comprising:

providing a total free counter for counting the number of free buffers in shared memory, the total free counter preset to the total number of buffers in the shared memory wherein the sum of the buffers in the multicast pool, the reserved pool and the shared pool is greater than the total number of buffers in the shared memory.

22. (Original) A switch as claimed in Claim 20 further comprising:  
providing a total free counter for counting the number of free buffers in shared memory, the total free counter preset to the total number of buffers in the shared memory wherein the sum of the buffers in the multicast pool, the reserved pool and the shared pool is equal to the total number of buffers in the shared memory.
- B' 23. (Original) A switch as claimed in Claim 21 further comprising the step of:  
determining the availability of a free buffer in the shared memory dependent on the number of free buffers in the shared memory stored in the total free counter.
24. (Original) A switch as claimed in Claim 22 further comprising the step of:  
determining the availability of a free buffer in the shared memory dependent on the number of free buffers in the shared memory stored in the total free counter.
-